次世代ロボット知能化技術開発プロジェクト

ロボット知能ソフトウェア再利用性向上技術の開発

来訪者受付システム
(RS002)

PA10 取扱説明書

0.5 版

2011 年 3 月 16 日

RTC 再利用技術研究センター

目次

1　はじめに

1.1　目的

本書は、三菱重工業製汎用ロボット PA10、シュンク・ジャパン社製電動式平行ハンド RH707、および産業技術総合研究所に提供して頂いた作業対象物認識モジュールを用いたハンドアイシステム(以降、PA10 システムと略す)の取扱方を記載した文書である。

1.2　本書での書式

本文書で使用している記号・書式の目的を下表に示す。

表 1-1 書式一覧

No.	記号･書式	目的
1	※	注意書き
2	赤色の文字	注記

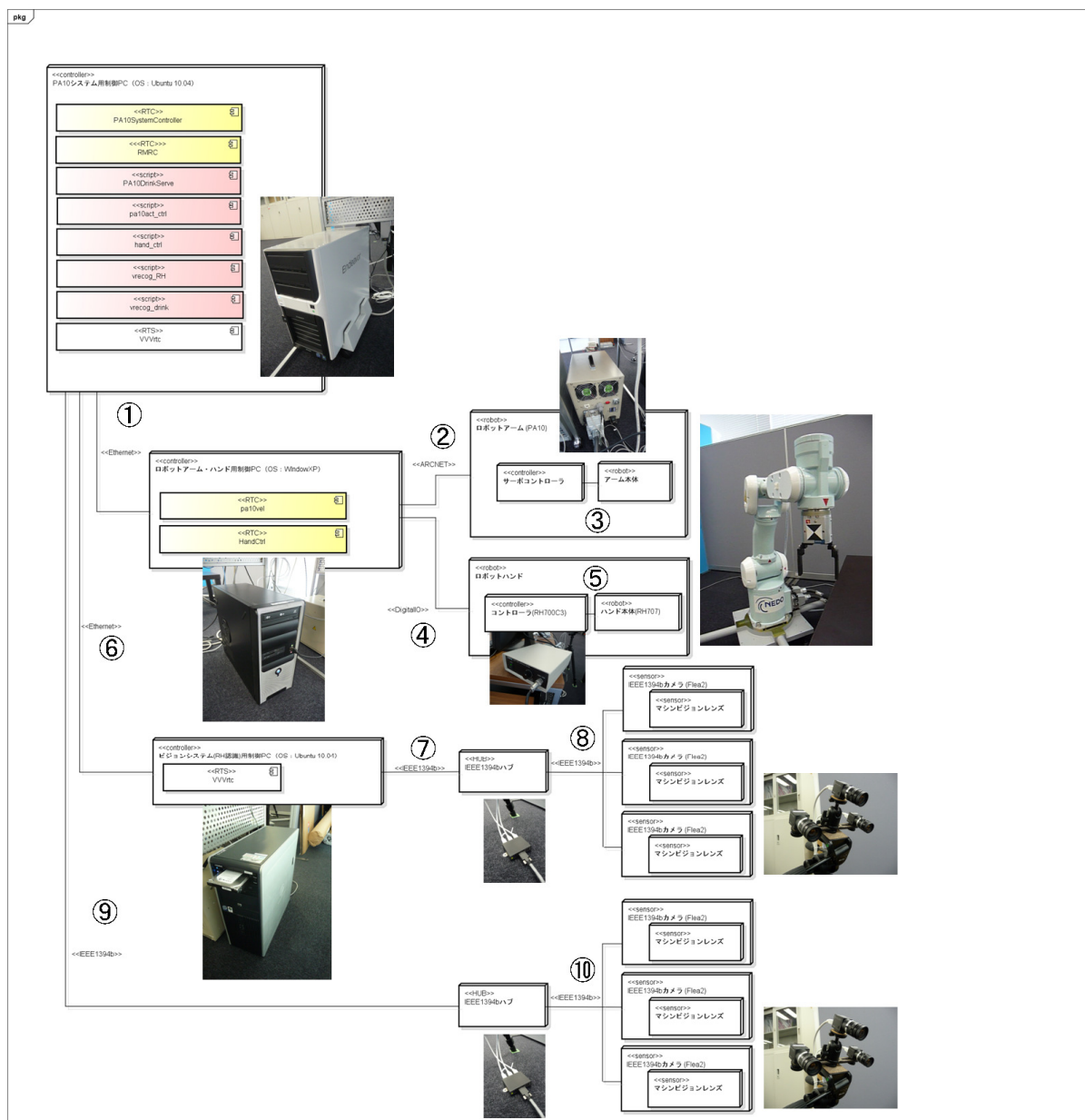
1.3　用語の定義、略語

表 1-2 用語の定義、略語一覧

No.	表記	意味
1	本システム	来訪者受付システム
2	プロジェクト	次世代ロボット知能化技術開発プロジェクト
3	センター	RTC再利用技術研究センター
4	現時点	本書作成時点(2010/10/01)
5	在籍者	センター内勤務者
6	OS	動作対象プラットフォーム
7	RTミドルウェア	OpenRTM-Aist
8	RTM	RTミドルウェア
9	RTC	RTコンポーネント
10	RTS	RTシステム
11	OSS	オープンソースソフトウェア
12	障害物	人及び、人が一人で運ぶ事の出来る物体
13	RH	リファレンスハードウェア
14	PA10システム	PA10、RH707および作業対象物認識モジュール群を用いたハンドアイシステム

1.4 参考資料

本書を作成するにあたり参照した文書・資料を下表に示す。

表 1-3 参考資料一覧

No.	文書名	備考／URL
1	OpenRTM-aist Official Website	http://www.is.aist.go.jp/rt/OpenRTM-aist/
2	RH707_機能仕様書	RTC再利用技術研究センターHP→再利用検証成果公開ページ→統合検証 成果報告（RS001）詳細ページ→参考資料 http://210.154.184.16/RS001/RS001.html
3	pa10vel_機能仕様書	
4	PA10RMRC_機能仕様書	
5	機能仕様書「作業対象物認識モジュール群」（産総研）	

2 システム概要

2.1 ハードウェア構成

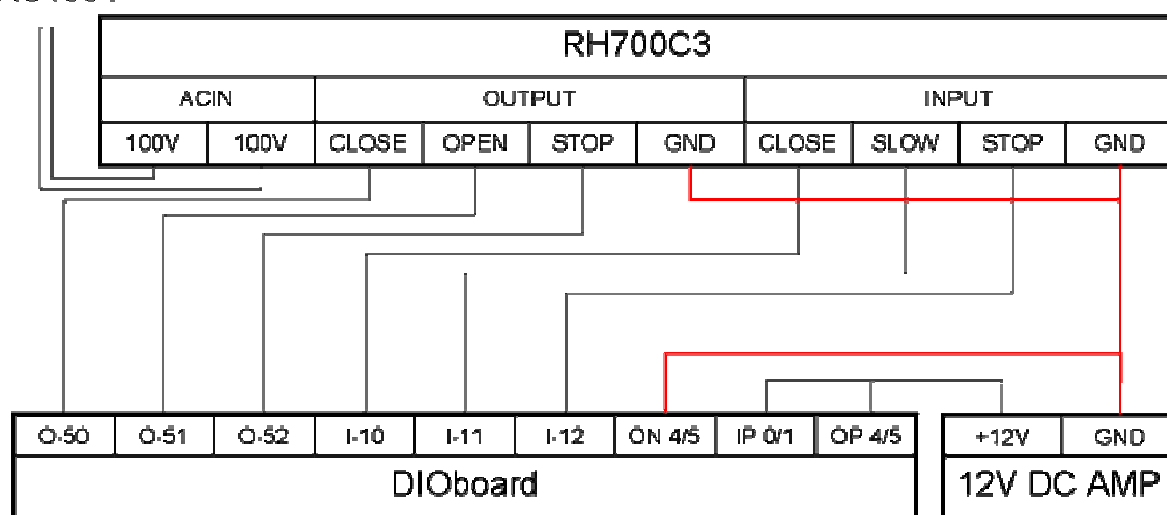
本システムのハードウェア構成について以下に記す。

図 2-1 ハードウェア構成図

本システムのハードウェア結線情報を以下に記す。

図 2-2 結線情報

番号	ケーブル名	接続元	接続先	備考
①	Ethernet	PA10 システム制御用 PC	ロボットアーム・ハンド用制御 PC	
②	ARCNET	ロボットアーム・ハンド用制御 PC（運動制御ボード）	サーボコントローラ	ケーブルは PA10 付属品
③	専用ケーブル	サーボコントローラ	ロボットアーム本体	
④	DigitalIO	ロボットアーム・ハンド用制御 PC（ディジタル入出力ボード）	ロボットハンドコントローラ	
⑤	専用ケーブル	ロボットハンドコントローラ	ロボットハンド本体	専用ケーブル
⑥	Ethernet	PA10 システム制御用 PC	ビジョンシステム（RH）制御用 PC	
⑦	IEEE1394b	ビジョンシステム（RH）制御用 PC	IEEE1394b ハブ	9 ピン－9 ピン
⑧	IEEE1394b	IEEE1394 ハブ	IEEE1394b カメラ	9 ピン－9 ピン
⑨	IEEE1394b	PA10 システム制御用 PC	IEEE1394b ハブ	9 ピン－9 ピン
⑩	IEEE1394b	IEEE1394 ハブ	IEEE1394b カメラ	9 ピン－9 ピン

図 2-3 RH700C3-PIO-32/32L(PCI)H 接続図

本システムのハードウェア仕様について以下に記す。

【PA10 システム用制御 PC】

デスクトップ PC

メーカー	EPSON
製品名	Endeavor Pro4350
仕様	CPU:Core2 Duo E8500 3.16GHz
	ChipSet:Intel P35 + ICH9R
	Memory:PC2-6400 4GB
	GPU:NVIDIA GeForce GTX260
	HDD:ST3250310AS 250GB SATA 7200rpm
OS	Ubuntu 10.04

【ロボットアーム・ハンド用制御 PC】

デスクトップ PC

メーカー	ライフロボティクス株式会社
製品名	RTコンポーネント開発用マシン
仕様	CPU:Core2 Duo E8400 3.00GHz
	Memory:PC2-6400 2GB
	GPU:
	HDD:HD251HJ 250GB
OS	Windows XP SP3

【ロボットアーム】

運動制御ボード

メーカー	三菱重工業株式会社
製品名	MHI-D7281
仕様	入力:PAライブラリによるコマンド発行(バス経由)
	出力:各軸の速度指令値(ARCNET経由)

サーボコントローラ

メーカー	三菱重工業株式会社
製品名	PA10-7C-CNT
仕様	関節数:7
	関節構成:ロボット取り付け面より R－P－R－P－R－P－R
	(R は回転関節，P は旋回関節を示す)
	関節名称:ロボット取り付け面よりS1－S2－S3－E1－E2－W1－W2
	(Sは肩関節，Eは肘関節，Wは手首関節を表す)
	アーム長　肩リーチ 317mm(ベース面～S2 間)
	上腕　450mm(S2～E1 軸間)
	下腕　480mm(E1～W1 軸間)
	手首リーチ　80mm(W1～メカニカルインタフェース面間)

ロボットアーム本体

メーカー	三菱重工業株式会社
製品名	PA10-7C-ARM
仕様	

【ロボットハンド】

デジタル入出力ボード

メーカー	CONTEC
製品名	PIO-32/32L(PCI)H
仕様	

ロボットハンドコントローラ

メーカー	シュンク・ジャパン株式会社
製品名	RH700C3
仕様	

ロボットハンド本体

メーカー	シュンク・ジャパン株式会社
製品名	RH707
仕様	

【ビジョンシステム(RH 認識)用制御 PC】

デスクトップ PC

メーカー	日本HP
製品名	HP Compaq Business Desktop dc7800
仕様	CPU:Core2 Quad Q9550 2.83GHz
	Memory:PC2-6400 4GB
	GPU:GeForce 7300 SE
	HDD:ST3250318AS 250GB 7200rpm
OS	Ubuntu 10.04

【カメラ】

IEEE1394b インターフェース

メーカー	Point Grey Research
製品名	DEVKIT-01-0002(Flea2開発キット)
仕様	

IEEE1394b ハブ（5Port）

メーカー	Point Grey Research
製品名	VP-FWB-HUB-5PORT-SET2
仕様	

IEEE1394b カメラ（×3）

メーカー	Point Grey Research
製品名	FL2-03S2M-C
仕様	

マシンビジョンレンズ（×3）

メーカー	FUJINON
製品名	DF6HA-1B
仕様	

3 準備

3.1 開発環境

1.PA10 システム用制御 PC

本 PC を用いて PA10 システムを統括的に制御する。
また、PA10 の分解運動速度制御や飲み物の認識も行う。

次節の手順に従い以下のソフトウェアをインストールする。

ｿﾌﾄｳｪｱ	OpenRTM-aist 1.0.0 (C++)
	OpenRTM-aist-Python 1.0.0
	python visual
	Eclipse関連ツール
	高機能3次元視覚システムVVV
RTC RTS ｽｸﾘﾌﾟﾄ	PA10SystemController
	RMRC
	作業対象物認識モジュール群
	各種制御ｽｸﾘﾌﾟﾄ

2. ビジョンシステム(RH 認識)用制御 PC

本 PC を用いてリファレンスハードウェアのマーカ（ドリンクホルダ）を認識し、リファレンスハードウェアが飲み物を受け取る位置に到着しているか否かを判定する。

次節の手順に従い以下のソフトウェアをインストールする。

ｿﾌﾄｳｪｱ	OpenRTM-aist 1.0.0 (C++)
	OpenRTM-aist-Python 1.0.0
	高機能3次元視覚システムVVV
RTS	作業対象物認識モジュール群

3. ロボットアーム・ハンド用制御 PC

本 PC を用いて PA10 実機制御、RH707 ハンドの制御を行う。

次節の手順に従い以下のソフトウェアをインストールする。

ｿﾌﾄｳｪｱ	OpenRTM-aist 1.0.0 (C++) 三菱重工製PAライブラリ コンテック社製 API関数ライブラリ集API-PAC(W32)
RTC	pa10vel HandCtrl

3.2 開発環境構築

3.2.1 PA10 システム用制御 PC

本 PC の OS は、**Ubuntu10.04** であるとし、以下の作業（**3.2.1.1 - 3.2.1.7** ）を行い環境を構築する。

3.2.1.1 制御 PC へのモジュールの導入

PA10 システムを構成するモジュールを下記 URL よりダウンロードする。
ダウンロードしたファイルには、以降にインストール手順を記載する OpenRTM に対するパッチファイル等も含まれているため、必ず本作業を初めに行うこと。

統合検証～ＲＳ００２～（ソースダウンロード）
ＵＲＬ：http://210.154.184.16/RS002/RS002.html

- ソース（全部）
- PA10 関連

モジュール	格納フォルダ	導入先ハードウェア
PCforSystemControl.zip	~/	PA10 システム用制御 PC
PCforVV.zip	~/	ビジョンシステム(RH 認識)制御 PC
device.zip	C:¥	ロボットアーム・ハンド用制御 PC
patch.zip	-	OpenRTM 用パッチ
1.0.0_Windows バグ対応.zip	-	Windows 版 OpenRTM バグ対応ファイル
VV_IDL	-	VV 関連 IDL ファイル

ファイルを解凍する。（ここでは~/にダウンロードしたものとする）
例）　「PA10 関連」をダウンロードした場合

```
unzip PA10.zip
cp ./PA10/PCforSystemControl.zip ./
unzip PCforSystemControl.zip
```

作業対象物認識モジュール群を利用する際には、対象物体に関する高機能 3 次元視覚システム VVV を必要とする。
詳細に関しては以下に問い合わせること。

<連絡先>
独立行政法人　産業技術総合研究所
知能システム研究部門　タスクビジョン研究グループ
河井　良浩
email: irtsp-vvv@ m.aist.go.jp

3.2.1.2 OpenRTM-aist-1.0.0 RELEASE (C++) Ubuntu 10.04 のインストール

C++言語で実装された RTC を実行するための RT ミドルウェアをインストールする。本項では、まず一括インストールスクリプトを使用し OpenRTM-aist-1.0.0-RELEASE とその依存パッケージの一括インストールを行う。その後 OpenRTM-aist-1.0.0-RELEASE のバグ対応を行い、再インストールする。

・設定方法
1.一括インストールスクリプト pkg_install100_ubuntu.sh を下記より適当なディレクトリ（ここでは~/）にダウンロードし、実行する。
※一括インストールスクリプトで「コンパイラや OmniORB4」をインストールする。
pkg_install100_ubuntu.sh ダウンロード
URL：http://www.openrtm.org/pub/OpenRTM-aist/cxx/install_scripts/

```
sudo sh ~/pkg_install100_ubuntu.sh
```

対応する問題

- InPort::read()でブロックされる問題 [openrtm-users 01308]参照
- Manager の shutdown に関連したバグ [openrtm-users 01149]参照
- ipv6 エントリのコメントアウト [openrtm-users 01270]参照

2. 下記よりソースファイルを（ここでは~/に）ダウンロードし展開する。

OpenRTM-aist-1.0.0-RELEASE.tar.gz ダウンロード
URL：http://www.openrtm.org/pub/OpenRTM-aist/cxx/1.0.0/

```
tar -xvzf OpenRTM-aist-1.0.0-RELEASE.tar.gz
```

3. **3.2.1.1** で導入した「PA10/patch」フォルダ内にあるパッチファイルを ~/OpenRTM-aist-1.0.0/src/lib/rtm 内にコピーする。

```
unzip ~/PA10/patch.zip
cp ~/PA10/patch/RingBuffer.h.diff ~/OpenRTM-aist-1.0.0/src/lib/rtm
cp ~/PA10/patch/Manager.cpp.diff ~/OpenRTM-aist-1.0.0/src/lib/rtm
```

4. パッチを当てる。

```
cd OpenRTM-aist-1.0.0/src/lib/rtm
patch < RingBuffer.h.diff
patch < Manager.cpp.diff
```

5. make する。

```
cd ~/OpenRTM-aist-1.0.0
~/OpenRTM-aist-1.0.0$./configure --prefix=/usr
~/OpenRTM-aist-1.0.0$make clean
~/OpenRTM-aist-1.0.0$make
~/OpenRTM-aist-1.0.0$sudo make install
```

6. /etc/hosts の localhost の ipv6 エントリをコメントアウトする。

```
# ::1        localhost ip6-localhost ip6-loopback
```

3.2.1.3 OpenRTM-aist-Python-1.0.0-RELEASE のインストール

Python 言語で実装された RTC を実行するための RT ミドルウェアをインストールする。Rtc_Handle を用いた RTC 操作が行えるよう、独自型データを Python 版 RTM モジュール群に追加する。

・設定方法

1. パッケージインストール

一括インストールスクリプト pkg_install_python_ubuntu.sh を下記から適当なディレクトリ（ここでは~/）にダウンロードし、実行する。

pkg_install_python_ubuntu.sh ダウンロード
ＵＲＬ：http://www.openrtm.org/OpenRTM-aist/download/install_scripts/

```
su
# sh pkg_install_python_ubuntu.sh
```

途中、いくつかの質問をたずねられるので、y あるいは Y を入力しながら完了させる。

2. VVVrtc の独自型データを Python 版 RTM モジュール群に追加する。
3.2.1.1 で導入した「PA10/VVV_IDL」フォルダ内にある VVV.idl を /usr/lib/python2.6/dist-packages/OpenRTM_aist/RTM_IDL にコピーする。

```
unzip ~/PA10/VVV_IDL.zip
sudo cp ~/PA10/VVV_IDL/VVV.idl /usr/lib/python2.6/dist-packages/OpenRTM_aist/RTM_IDL
```

3.IDL コンパイルをしなおす

```
cd /usr/lib/python2.6/dist-packages/OpenRTM_aist/RTM_IDL
/usr/lib/python2.6/dist-packages/OpenRTM_aist/RTM_IDL$sudo omniidl -bpython VVV.idl
```

3.2.1.4 Eclipse 関連ツール のインストール

RTC を接続したり、状態を監視するためのツール RTSystemEditor をインストールする。RTSystemEditor は Eclipse のプラグインとして提供されるものであるため Eclipse もインストールする必要がある。

以下の URL から RTSystemEditor をプラグインした Eclipse（Linux 用全部入り）をダウンロードしインストールする。

全部入りパッケージ
ＵＲＬ：http://www.openrtm.org/openrtm/ja/node/941#package
また上記参照先に記載されている Eclipse 動作不具合への対応も必ず行う。

3.2.1.5 python visual のインストール

シミュレータ環境として、python visual をインストールする。

1. Synaptic マネージャを起動する。
「システム」→「システム管理」→「Synaptic パッケージ・マネージャ 」と選択する。

2. python visual をインストールする。
「python visual」で検索し、python-visual にチェックを入れ、適用する。

3.「libgtkglext」で検索し、libgtkglext1 と libgtkglext1-dev にチェックを入れ、適用する。

3.2.1.6 RTC のコンパイル

すでに **3.2.1.1** で導入した RTC のバイナリファイルは、/PCforSystemControl/bin/comp 内に同封済みであるが、これらのコンパイルは以下の様にしてできる。

- PA10SystemController

```
cd ~/PCforSystemControl/src/GeneralController/PA10SystemController
~/PCforSystemControl/src/GeneralController/PA10SystemController$make -f
Makefile.PA10SystemController
~/PCforSystemControl/src/GeneralController/PA10SystemController$make -f
Makefile.PA10SystemController install
```

- RMRC、vel_7dof、pa10disp、pa10hand_disp、slider

```
cd ~/PCforSystemControl/src/MotionController
~/PCforSystemControl/src/MotionController$makeall.sh
```

3.2.1.7 設定ファイル修正

~/PCforSystemControl/etc/set_env.py にあるネームサーバーを適宜修正する。

```
#!/usr/bin/env python
# -*- Python -*-import subprocess
#
          :
#Nameserver
RMRC_NS     = "localhost:9876"
DEVICE_NS = "○○○○:9876"
DRINK_NS    = "localhost:5005"
RH_NS        = "○○○○:9876"
PACT_NS     = "localhost:2809"
CT_NS        = "○○○○:2809"

  ○○○○に使用する PC の IP アドレスを記述する
```

3.2.2 ビジョンシステム(RH 認識)用制御 PC

本 PC の OS は、**Ubuntu10.04** であるとし、以下の作業（**3.2.2.1 - 3.2.2.3**）を行い環境を構築する。

3.2.2.1 制御 PC へのモジュールの導入

PA10 システムを構成するモジュールを下記 URL よりダウンロードする。ダウンロードしたファイルには、以降にインストール手順を記載する OpenRTM に対するパッチファイル等も含まれているため、必ず本作業を初めに行うこと。

統合検証～ＲＳ００２～（ソースダウンロード）
ＵＲＬ：http://210.154.184.16/RS002/RS002.html

- ソース（全部）

- PA10 関連

モジュール	格納フォルダ	導入先ハードウェア
PCforSystemControl.zip	~/	PA10 システム用制御 PC
PCforVVV.zip	~/	ビジョンシステム(RH 認識)制御 PC
device.zip	C:¥	ロボットアーム・ハンド用制御 PC
patch.zip	-	OpenRTM 用パッチ
1.0.0_Windows バグ対応.zip	-	Windows 版 OpenRTM バグ対応ファイル
VVV_IDL	-	VVV 関連 IDL ファイル

ファイルを解凍する。（ここでは~/にダウンロードしたものとする）

例）「PA10 関連」をダウンロードした場合

```
unzip PA10.zip
cp ./PA10/PCforVVV.zip ./
unzip PCforVVV
```

作業対象物認識モジュール群を利用する際には、対象物体に関する高機能 3 次元視覚システム VVV を必要とする。詳細に関しては以下に問い合わせること。

<連絡先>
独立行政法人 産業技術総合研究所
知能システム研究部門 タスクビジョン研究グループ
河井 良浩
email: irtsp-vvv@ m.aist.go.jp

3.2.2.2 OpenRTM-aist-1.0.0 RELEASE (C++) Ubuntu 10.04 のインストール

PA10 システム用制御 PC のときと同様に行う。

3.2.2.3 python visual のインストール

PA10 システム用制御 PC のときと同様に行う。

3.2.3 ロボットアーム・ハンド用制御 PC

本 PC の OS は、Windows XP であるとし、以下の作業（**3.2.3.1 - 3.2.3.5** ）を行い環境を構築する。

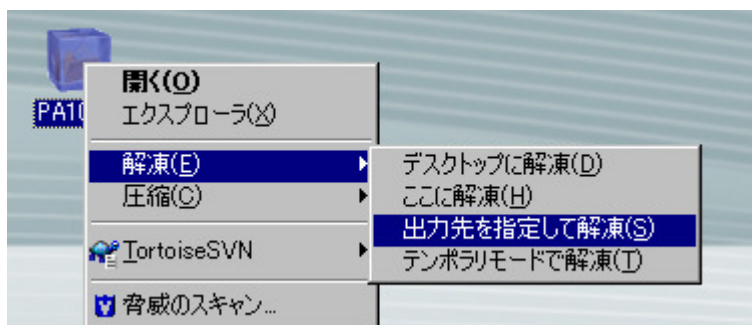
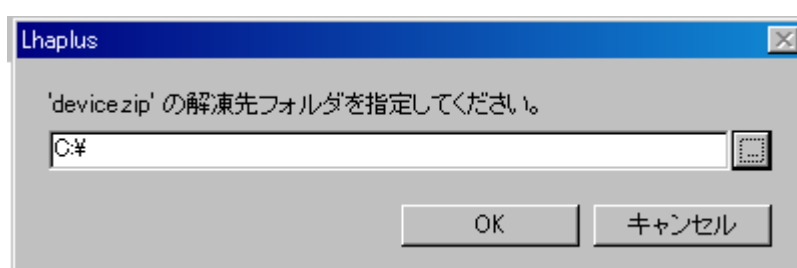
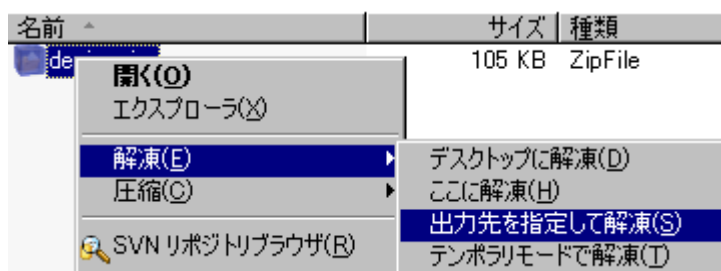
3.2.3.1 制御 PC へのモジュールの導入

PA10 システムを構成するモジュールを下記 URL よりダウンロードする。
ダウンロードしたファイルには、以降にインストール手順を記載する OpenRTM に対するパッチファイル等も含まれているため、必ず本作業を初めに行うこと。

統合検証～ＲＳ００２～（ソースダウンロード）
ＵＲＬ：http://210.154.184.16/RS002/RS002.html

- ソース（全部）
- PA10 関連

モジュール	格納フォルダ	導入先ハードウェア
PCforSystemControl.zip	~/	PA10 システム用制御 PC
PCforVVV.zip	~/	ビジョンシステム(RH 認識)制御 PC
device.zip	C:¥	ロボットアーム・ハンド用制御 PC
patch.zip	-	OpenRTM 用パッチ
1.0.0_Windows バグ対応.zip	-	Windows 版 OpenRTM バグ対応ファイル
VVV_IDL	-	VVV 関連 IDL ファイル

ファイルを C ドライブ直下（C:¥）に解凍する。（zip ファイル解凍環境は各自用意するものとする。）
例） 「PA10 関連」をダウンロードした場合
「PA10.zip」をデスクトップへ解凍する。

さらにデスクトップに作成された「PA10」というフォルダ内にある「device.zip」を「C：￥」へ解凍する。

3.2.3.2 OpenRTM-aist-1.0.0-RELEASE (C++) Windows のインストール

C++言語で実装された RTC を実行するための RT ミドルウェアをインストールし、バグ対応を行う。

1. 下記の URL を参考にして OpenRTM-aist-1.0.0-RELEASE (C++) Windows をインストールする。
URL: http://www.openrtm.org/openrtm/ja/content/windows%E3%81%B8%E3%81%AE%E3%82%A4%E3%83%B3%E3%82%B9%E3%83%88%E3%83%BC%E3%83%AB

Ubuntu 10.04 版のものと同様のバグ対応を行う。
2. **3.2.1.1** で導入した「PA10」フォルダ内にある 1.0.0_Windows バグ対応.zip ファイルを解凍する。

3. フォルダ内ファイルを、C:\Program Files\OpenRTM-aist\1.0\bin にコピー(上書き)する。

3.2.3.3 三菱重工業製 PA ライブラリ のインストール

PA10 の制御には三菱重工業製 PA ライブラリ（商用）を使用する。ライブラリのインスト

ールや運動制御ボードの設置､機器接続については付属のソフトウェアインストールマニュアルに従う。

尚、ソフトウェアインストール先は以下の通りとする。

C:¥WinPApci

3.2.3.4 コンテック社製 API 関数ライブラリ集 API-PAC(W32) のインストール

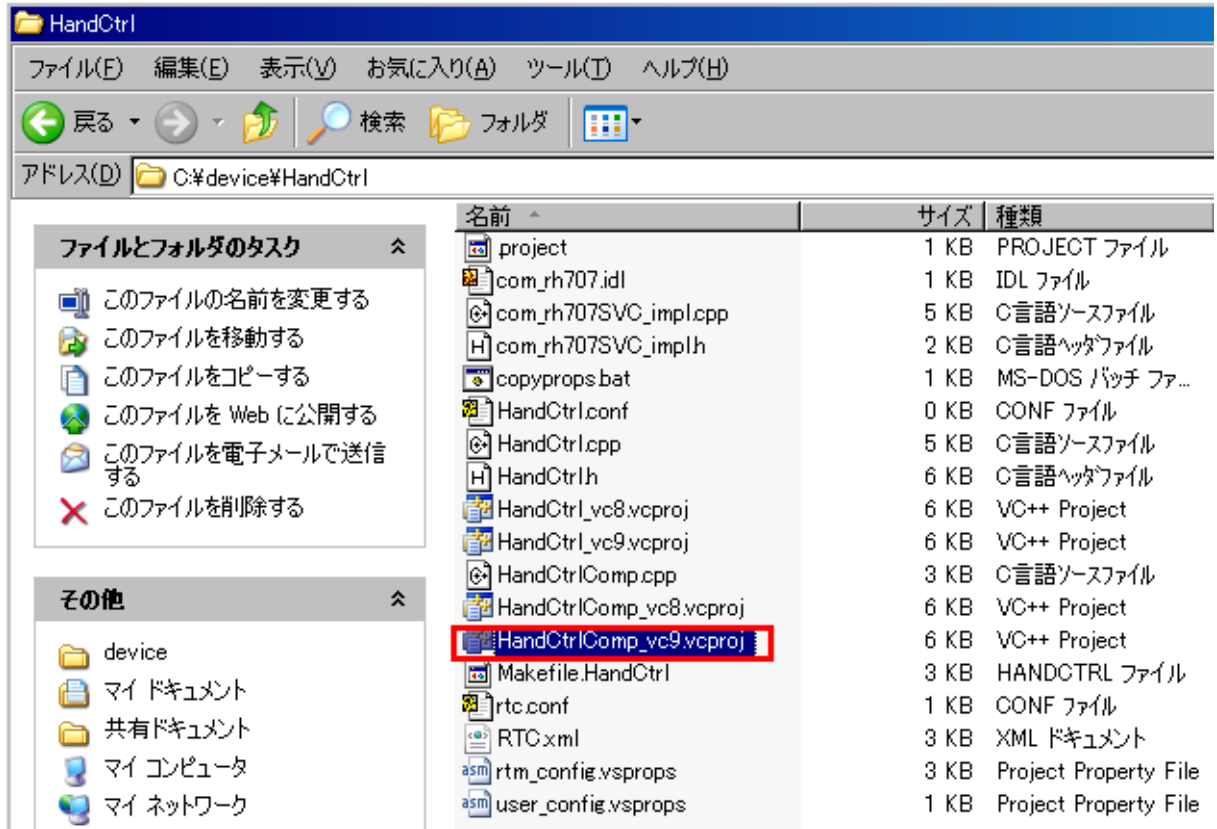
RH707 の制御にはコンテック社製 API 関数ライブラリ集 API-PAC(W32)（商用）を使用する。ライブラリのインストールや運動制御ボードの設置、機器接続については付属の説明書に従う。

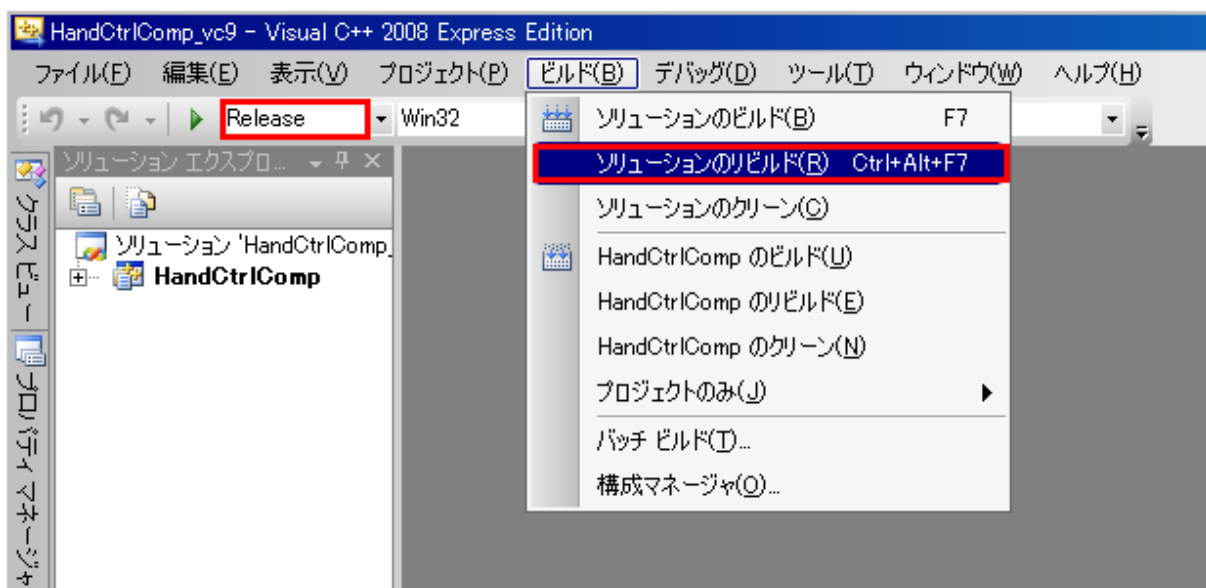
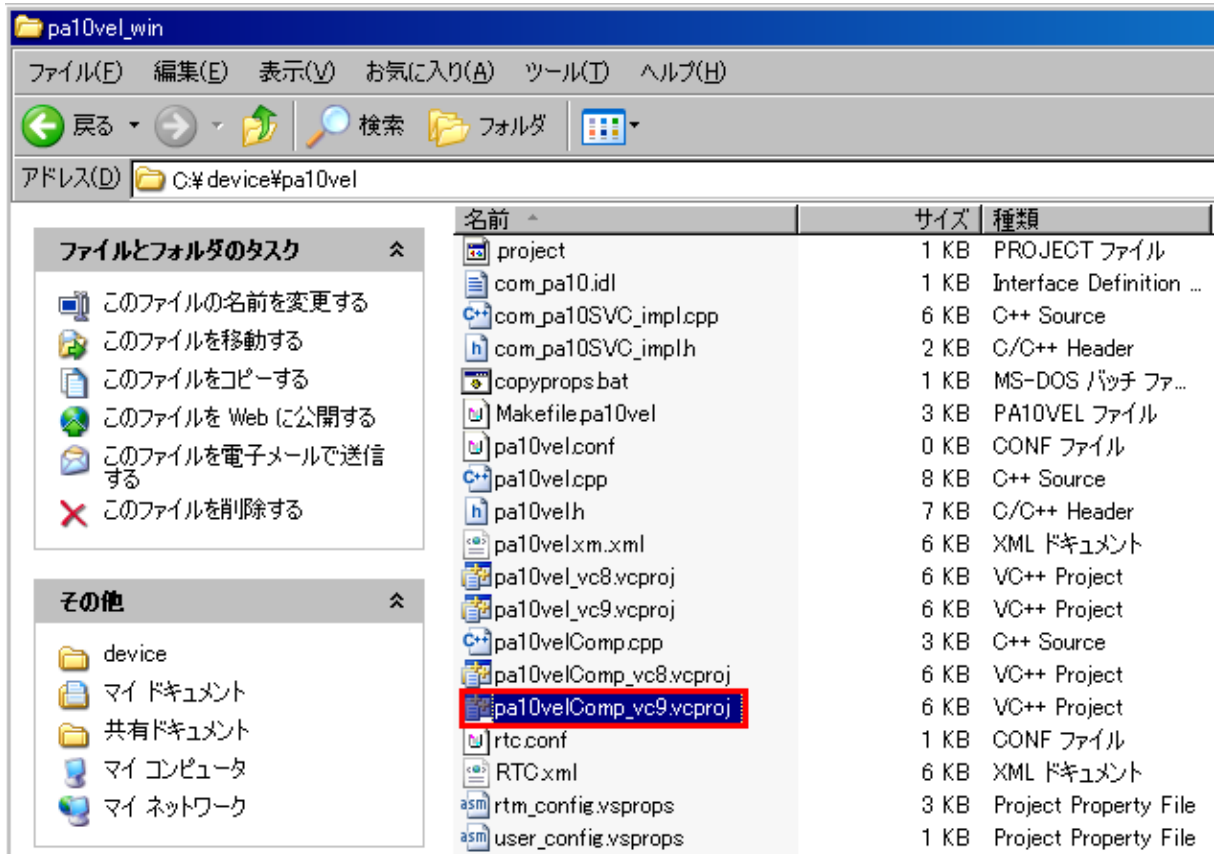
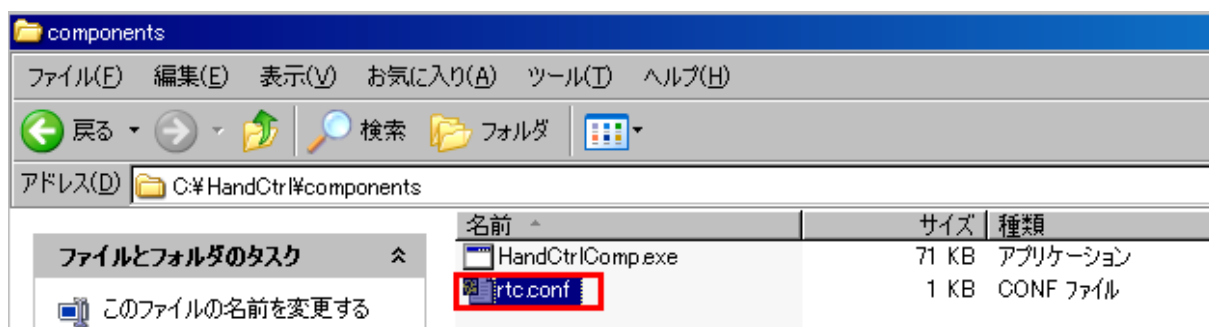
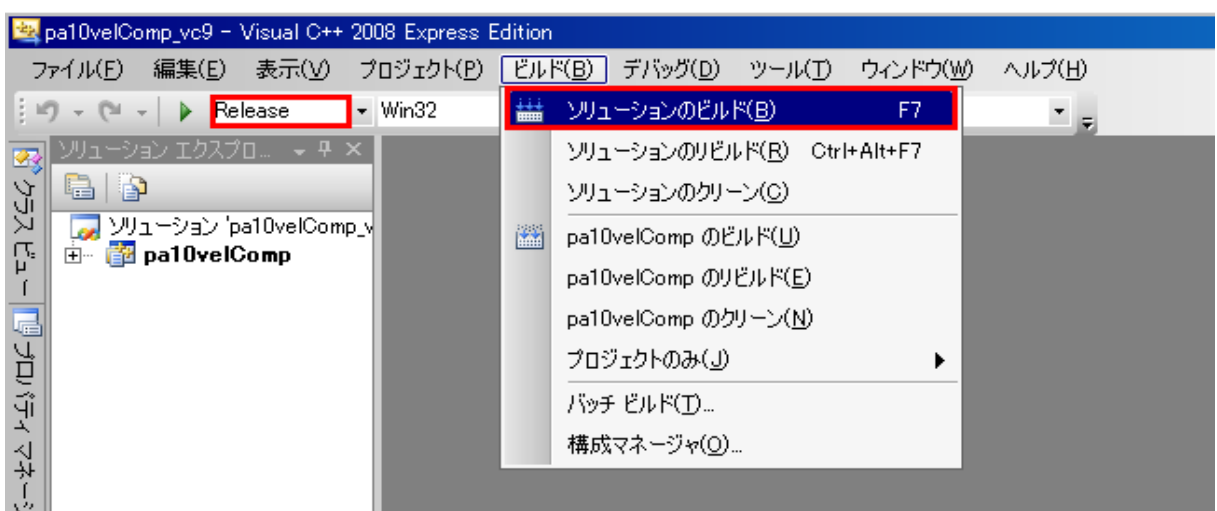
尚、ソフトウェアインストール先は以下の通りとする。

C:¥CONTEC

3.2.3.5 RTC のコンパイル

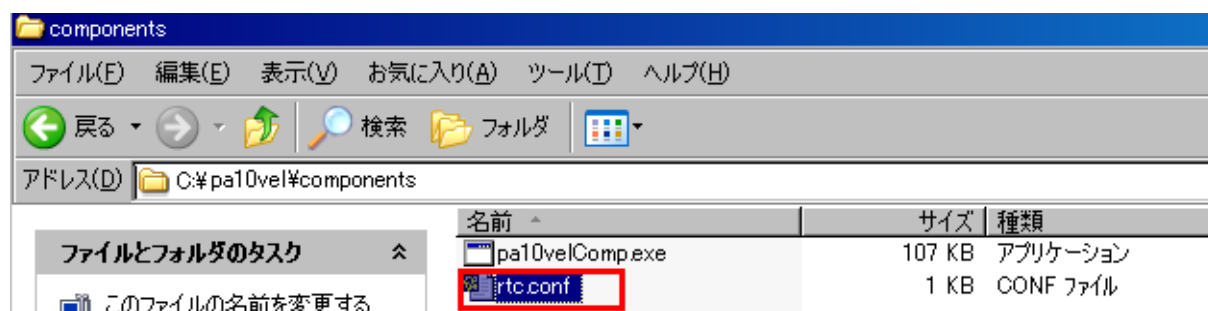
ソースをビルドする。

- HandCtrl

C:¥device¥HandCtrl フォルダ内にある HandCtrlComp_vc9.vcproj を開く。

「Release」モードに切り替え、「ビルド」タブから「ソリューションのビルド」を選択し実行する。

C:￥device￥HandCtrl フォルダ内に components というフォルダが作成されていることを確認し、その中に C:￥device￥HandCtrl￥rtc.conf をコピーする。

- pa10vel

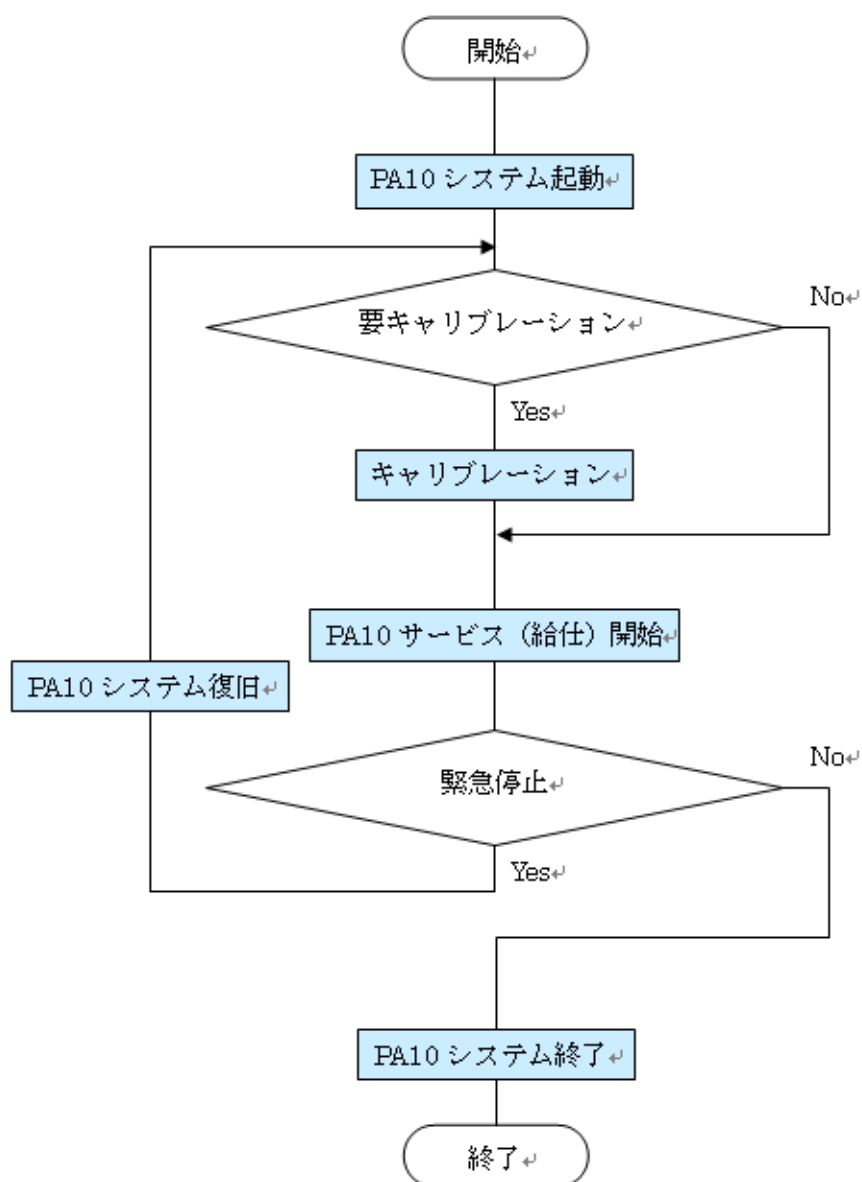
C:￥device￥pa10vel フォルダ内にある pa10velComp_vc9.vcproj を開く。

「Release」モードに切り替え、「ビルド」タブから「ソリューションのビルド」を選択し実行する。

C:￥device￥pa10vel フォルダ内に components というフォルダが作成されていることを確認し、その中に C:￥device￥pa10vel￥rtc.conf をコピーする。

4　使用方法

本章では、PA10 システムを来訪者受付システムの一部として使用するための方法と、
来訪者受付システムとは独立に PA10 システムの機能を単体で操作（モジュール単体操作）するための方法の２つについて記載する。後者は PA10 システムの機能毎の動作確認を行いたい場合等に使用する。

4.1　PA10 システム

PA10 システムの使用方法を以下に示す。

4.1.1 PA10 システム起動

1. PA10 の周りにあるもの（テーブル等）をどける。
2. サーボコントローラの電源を入れ、非常停止ボタンを OFF にする。
3. PA10 システム用制御 PC、ロボットアーム・ハンド用制御 PC、ビジョンシステム(RH 認識)用制御 PC の電源を入れる。
4. Eclipse を起動する。

```
cd eclipse
~/eclipse$sh eclipse.sh      ←Eclipse 動作不具合対応として作成した sh
```

URL:　http://www.openrtm.org/openrtm/ja/node/941#toc0

5. PA10 の起動。

5.1 PA10 システム用制御 PC 上でネームサーバ（rtm-naming 9876）を立ち上げる。

```
rtm-naming 9876
```

5.2. PA10 システム用制御 PC 上でネームサーバ（rtm-naming 2809）を立ち上げる。

```
rtm-naming 2809
```

5.3. PA10 システム用制御 PC 上で PA10 システム制御 RTC を起動する。

```
cd PCforSystemControl/bin/comp
~/PCforSystemControl/bin/comp$./PA10SystemControllerComp
-f ../../etc/PA10SystemController.conf
```

5.4. PA10 システム用制御 PC 上で分解運動速度制御 RTC を起動する。

```
cd PCforSystemControl/bin/script
~/PCforSystemControl/bin/script$sh PA10SysRun.sh
```

5.5. ロボットアーム・ハンド用制御 PC でネームサーバ（rtm-naming 9876）を立ち上げる。

```
rtm-naming 9876
```

5.6. ロボットアーム・ハンド用制御 PC で PA10 実機制御 RTC を起動する。

RTC を実行した際に PA10 が待機姿勢に移行するので注意する。

```
C:¥Documents and Settings¥rtc-center>cd ¥PA10¥device¥pa10vel_win¥components
C:¥device¥pa10vel_win¥components>pa10velComp.exe
```

6. RH707 ハンドを起動する。

6.1. ロボットハンドコントローラの電源を入れる。

6.2. ロボットアーム・ハンド用制御 PC で RH707 制御 RTC を立ち上げる。

```
C:¥Documents and Settings¥rtc-center>cd C:¥device¥pa10vel¥components
C:¥device¥pa10vel¥components>pa10velComp.exe
```

7. カメラを起動する。

7.1. PA10 システム用制御 PC 上でネームサーバ（rtm-naming 5005）を立ち上げる。

```
rtm-naming 5005
```

7.2. PA10 システム用制御 PC 上で作業対象物認識 RTC 群（飲み物認識用）を立ち上げる。

```
cd PCforSystemControl/bin/comp
~/PCforSystemControl/bin/script$sh start_comp2.sh
```

7.3. ビジョンシステム(RH 認識)用制御 PC でネームサーバ（rtm-naming 9876）を立ち上げる。

```
rtm-naming 9876
```

7.4. ビジョンシステム(RH 認識)用制御 PC で作業対象物認識 RTC 群（RH 認識用）を立ち上げる。

```
cd PCforVVV
~/PCforVVV$sh start_comp2.sh
```

4.1.2 キャリブレーション

PA10 システムにおいて、対象物の座標（PA10 ベース座標系）を取得するためにカメラ座標キャリブレーションを行う。キャリブレーションは 14 点のサンプルデータを基にして行われ、各サンプル点座標は、PA10 手先につけたクロスマークをカメラで認識させ取得する。キャリブレーションは高機能 3 次元視覚システムにおける操作手順に従って行うものとする。

1. PA10 システム用制御 PC 上でキャリブレーション用 PA10 動作スクリプトを実行する。

```
cd PCforSystemControl/bin/script
~/PCforSystemControl/bin/script$python vvv_calib.py
                :
>>>need ready(1)?    :y or n
```

"y"と入力する。

```
>>>need ready(1)?    :y or n    y
          :
>>>Select mode    [ table: 1     RH: 2     QUIT: 3]    :
```

飲み物認識用カメラの場合は 1、RH 認識用カメラの場合は 2 と入力する。

ex. 飲み物認識用カメラの場合

```
>>>Select mode    [ table: 1     RH: 2     QUIT: 3]    :1
```

PA10 の手先がキャリブレーション開始位置へ移動する。
コンソールには、

```
>>>Select mode    [ table: 1     RH: 2     QUIT: 3]    :1
                 :
        move to 1..    OK?
```

と出力され、キー入力待ちになる。以降、" Enter キー "を入力することで手先を順次キャリブレーション点へ移動させることができる。

2. カメラ座標キャリブレーション開始

- 飲み物認識カメラのキャリブレーションは PA10 システム用制御 PC 上で行う。
- RH 認識カメラのキャリブレーションはビジョンシステム(RH 認識)用制御 PC 上で行う。

3. キャリブレーション用 PA10 動作スクリプトを終了する。

```
>>>Select mode    [ table: 1      RH: 2      QUIT: 3]    :3
```

4.1.3 PA10 サービス（仕様）開始

1. テーブルを設置する。
   参考：PA10 動作仕様書 4.1.5.1 飲み物搭載配置図
2. テーブルの上に必要なだけお茶を設置する。

図 4-1 お茶設置図

飲み物はカメラから見て重ならないように設置するのが望ましい。カメラ取得画像の確認は作業対象物認識モジュールの単体操作（4.2.2）にて行うことができる。

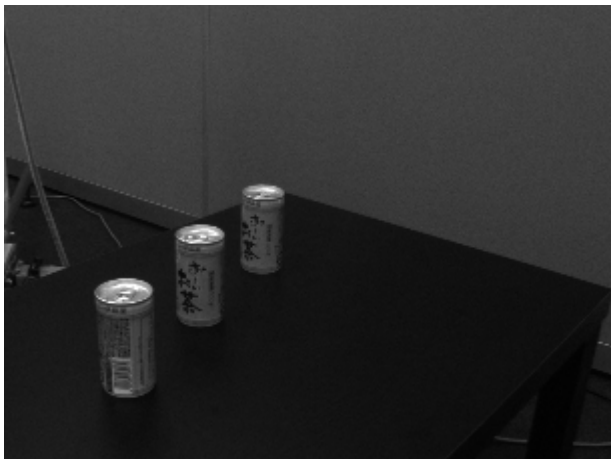
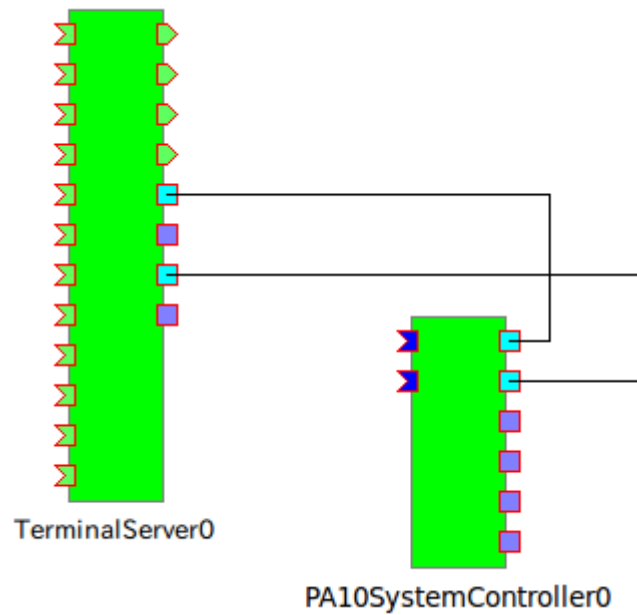
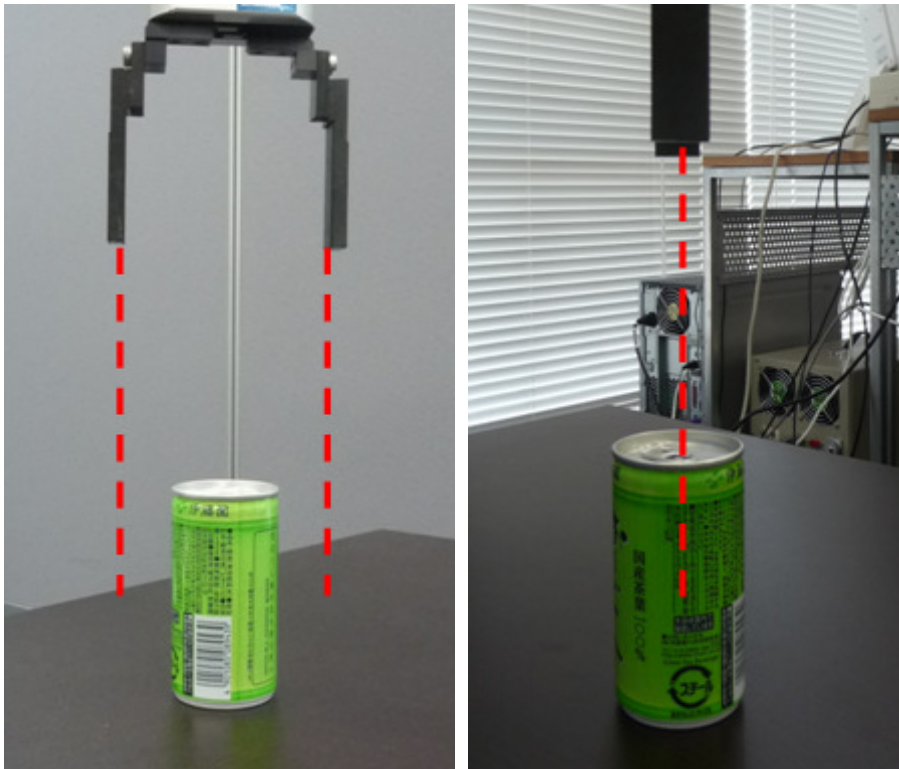
図 4-2　カメラ画像が取得する画像

3. PA10 システム起動時に立ち上げた RTSystemEditor 上で制御端末 RTC と PA10 システム制御 RTC のポートを接続し、PA10 システム制御 RTC を活性化させる（ポート接続図参照）。

図 4-3　ポート接続図

以上を終えると、制御端末からの要求に従い PA10 が給仕を開始する。給仕要求が発生するまでは PA10 は待機状態となる。

4. 給仕中の操作

制御端末から給仕の指示があると、PA10 システムによる RH への飲み物給仕が始まる。

PA10 の手先が目標の真上に来ている（まっすぐ下降してもツメと飲み物、もしくは把持した飲み物とドリンクホルダが干渉しない）かどうかを目視で確認しながら行う。

把持前確認

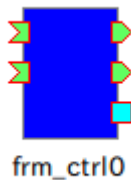
ツメの軌跡が赤い破線の通りになりそうならば OK とする。

図 4-4 把持前確認

搭載前確認

缶の底がホルダの穴（赤い実線）の真上に来ていれば OK とする。

図 4-5 搭載前確認

4.1.4 PA10 システム終了

1. PA10 システム起動時に立ち上げた RTSystemEditor 上で PA10SystemController を非活性化させる。この作業により RMRC および VVV 関連 RTC が終了する。

2. ロボットアーム・ハンド用制御 PC で PA10 実機制御 RTC を終了する。
PA10 実機制御 RTC が立ち上がっているコンソールをアクティブにして、Ctrl+c を押す。
このとき PA10 が直立姿勢へ戻るので注意すること。

3. ロボットアーム･ハンド用制御 PC で RH707 制御 RTC を終了する。RH707 制御 RTC が立ち上がっているコンソールをアクティブにして、Ctrl+c を押す。

4. PA10 システム用制御 PC 上で PA10SystemController を終了する。
PA10 システム用制御 PC 上で PA10SystemController が立ち上がっているコンソールをアクティブにして、Ctrl+C を押す。

4.1.5 PA10 システム復旧

1. Rtc_Handle でハンドリングしていた RTC を全て終了する。（ PA10 システム終了　を参照）
ハンドリングしたままになっている RTC が残っていると、以降、二重にハンドリングすることになってしまうため良くない。ハンドリングしたままの RTC は、下図に示される通り RTSystemEditor 上でポート間を線が結んでいなくてもポートが接続時の色で示されている。

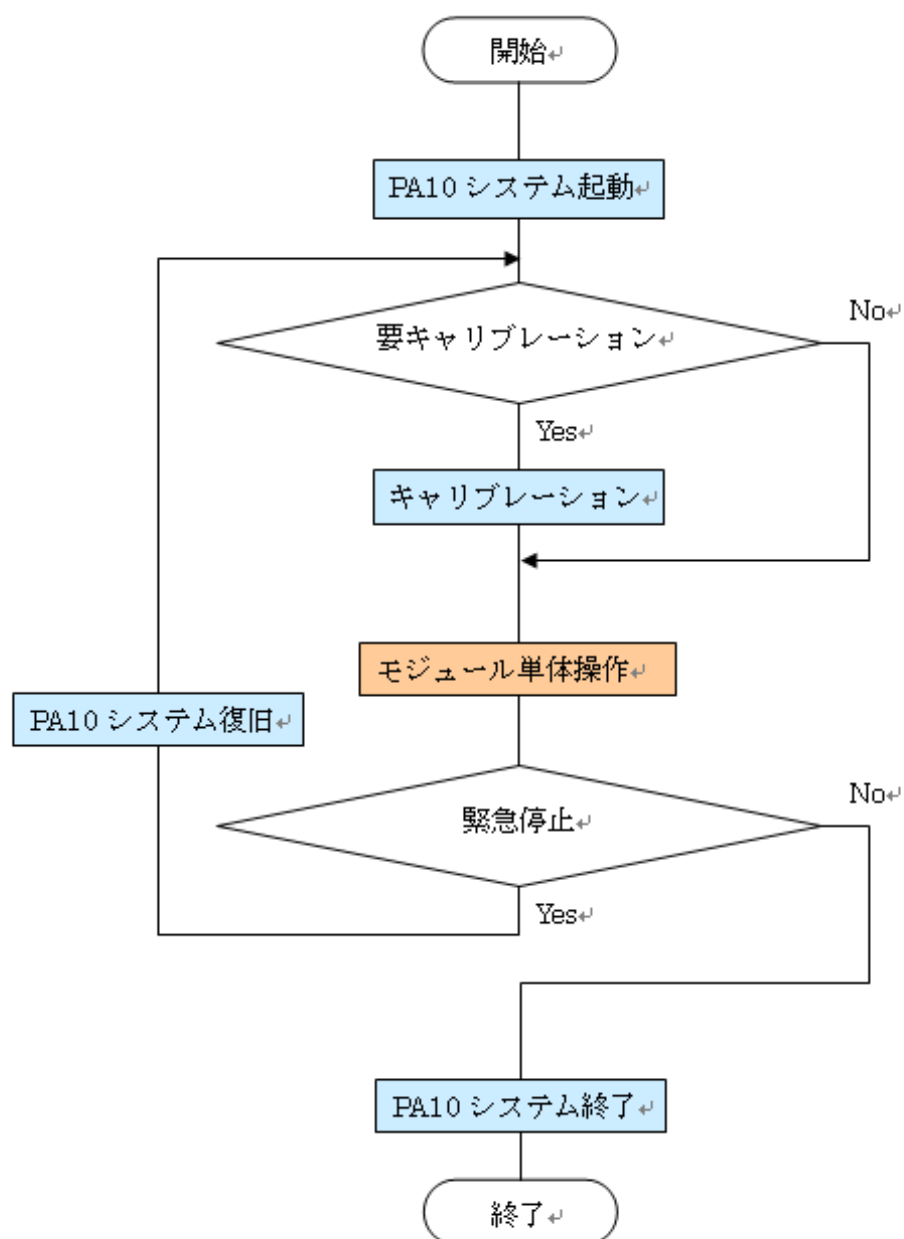
図 4-6 ハンドリングしたままの RTC

2. 非常停止によって実行不能になってしまった python スクリプトを終了させる。
スクリプト実行コンソールをアクティブにし、Ctrl+d を押す。

3. 非常停止ボタンを OFF にする。

1.で終了した RTC を再度起動する。　（ PA10 システム起動 を参照）

4.2 モジュール単体操作

以下のように各制御スクリプトを実行することで、モジュール毎の動作確認を行うことができる。

4.2.1 PA10 単体操作

- 実機操作

1. PA10 実機制御スクリプトをインタラクティブモードで実行する。

```
cd PCforSystemControl/bin/script
~/PCforSystemControl/bin/script$python
```

pa10act_ctrl をインポートする。

```
          :
>>>from pa10act_ctrl    import *
```

RTC 間でのポート接続や Activate が行われ、実機が待機姿勢へ移行する。
以降、インタラクティブモードで PA10 実機制御が可能になる。制御コマンドについては以下に記載してある。
参考：PA10 詳細設計書 4.3. script 内関数（コマンド） - PA10 制御スクリプト

2. PA10 実機制御スクリプトを終了する。
end()関数を実行し、PA10 実機制御スクリプトを終了させる。

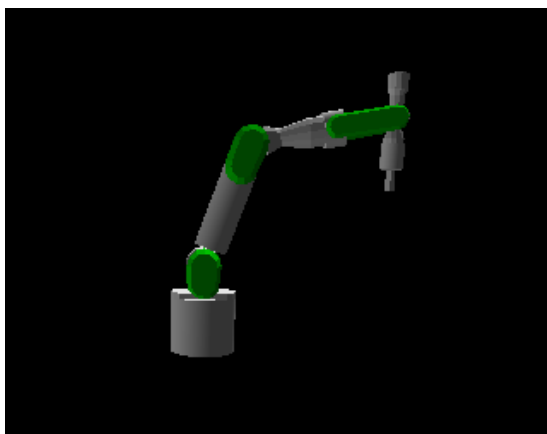
```
>>>end(env)
```

Ctrl+d を押し、python インタラクティブモードを終了する。

- シミュレータ環境での操作

1. PA10 システム用制御 PC 上で必要な RTC を起動する。

```
cd PCforSystemControl/bin/script
~/PCforSystemControl/bin/script$sh killPA10rtc.sh
~/PCforSystemControl/bin/script$sh simPA10SysRun.sh
```

2. シミュレーション用 PA10 制御スクリプトをインタラクティブモードで実行する。

```
cd PCforSystemControl/bin/script
~/PCforSystemControl/bin/script$python
```

pa10sim_ctrl をインポートする。

```
                  :
>>>from pa10sim_ctrl    import *
```

RTC 間でのポート接続や Activate が行われ、シミュレータ上の PA10 モデルが待機姿勢へ移行する。

図 4-7 シミュレータ上の PA10 モデル

以降、インタラクティブモードで PA10 のシミュレーションが可能になる。制御コマンドについては以下に記載してある。

参考：PA10 詳細設計書 4.3. script 内関数（コマンド） - PA10 制御スクリプト

3. シミュレーション用 PA10 制御スクリプトを終了する。

end()関数を実行し、シミュレーション用 PA10 制御スクリプトを終了させる。

```
>>>end(env)
```

Ctrl+d を押し、python インタラクティブモードを終了する。

4. シミュレーション用の RTC を終了させる場合は以下を実行する。

```
cd PCforSystemControl/bin/script
~/PCforSystemControl/bin/script$sh killPA10rtc.sh
```

4.2.2 作業対象物認識 RTC 群単体操作

- 飲み物認識用カメラの場合

1. カメラ視野内に認識可能なもの（ex. 190mm 缶）を置く。
2. 飲み物認識スクリプトをインタラクティブモードで実行する。

```
cd PCforSystemControl/bin/script
~/PCforSystemControl/bin/script$python
```

vrecog_drink をインポートする。

```
        :
>>>from vrecog_drink    import *
```

RTC 間のポート接続や Activate が行われる。以降、インタラクティブモードで物体認識が可能になる。制御コマンドについては以下に記載してある。
参考：PA10 詳細設計書 4.3. script 内関数（コマンド） - 飲み物認識スクリプト

3. 飲み物認識スクリプトを終了する。

end()関数を実行し、飲み物認識スクリプトを終了させる。

```
>>>end(env)
```

Ctrl+d を押し、python インタラクティブモードを終了する。

- RH 認識用カメラの場合

1. カメラ視野内に認識可能なもの（ex. ドリンクホルダ）を置く。
2. RH 認識スクリプトをインタラクティブモードで実行する。

```
cd PCforSystemControl/bin/script
~/PCforSystemControl/bin/script$python
```

vrecog_RH をインポートする。

```
        :
>>>from vrecog_RH    import *
```

RTC 間のポート接続や Activate が行われる。以降、インタラクティブモードで物体認識が可能になる。制御コマンドについては以下に記載してある。
参考：PA10 詳細設計書 4.3. script 内関数（コマンド） - RH 認識スクリプト

3. RH 認識スクリプトを終了する。
end()関数を実行し、RH 認識スクリプトを終了させる。

```
>>>end(env)
```

Ctrl+d を押し、python インタラクティブモードを終了する。

4.2.3 RH707 単体操作

1. ハンド制御スクリプトをインタラクティブモードで実行する。

```
cd PCforSystemControl/bin/script
~/PCforSystemControl/bin/script$python
```

hand_ctrl をインポートする。

```
        :
>>>from hand_ctrl    import *
```

RTC 間のポート接続や Activate が行われる。以降、インタラクティブモードで RH707 の制御が可能になる。制御コマンドについては以下に記載してある。
参考：PA10 詳細設計書 4.3. script 内関数（コマンド） - ハンド制御スクリプト

2. ハンド制御スクリプトを終了する。
deactivate()関数を実行し、ハンド制御スクリプトを終了させる。

```
>>>deacivate()
```

Ctrl+d を押し、python インタラクティブモードを終了する。

4.2.4 PA10 サービス（給仕）シミュレーション

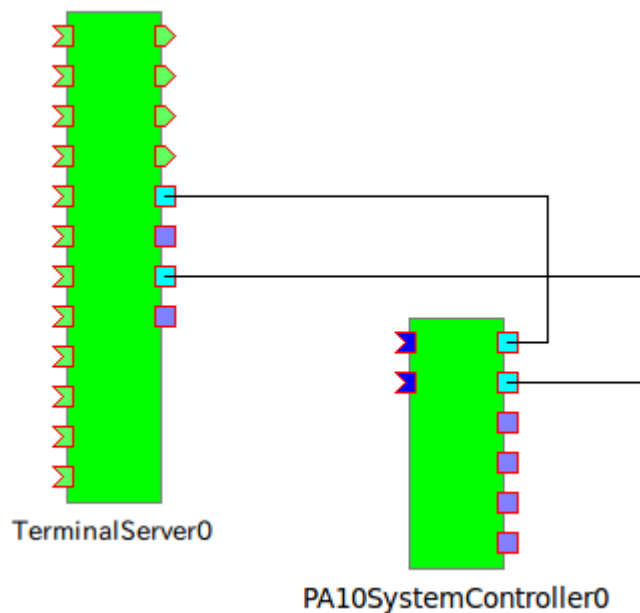
PA10 サービス（給仕）の実機動作をシミュレーションする場合は以下の様にする。
1. PA10 システム起動の手順 3.から 5.3.までを行う。
2. PA10 システム起動の手順 7.を行う。
3. シミュレーションに必要な RTC を起動する。

```
cd PCforSystemControl/bin/script
~/PCforSystemControl/bin/script$sh killPA10rtc.sh
~/PCforSystemControl/bin/script$sh simPA10SysRun.sh
```

4. PA10 システム起動の手順 1.から 2.までを行う。

5. PA10 システム起動時に立ち上げた RTSystemEditor 上で、PA10SystemController のコンフィギュレーションセットを変更する。

コンフィグレーション名	認識カメラの使用	アーム・ハンドの使用
ServiceSimulation_dmydata	なし（ダミーの認識結果データを使用）	VPython を用いたシミュレータモデル
ServiceSimulation	あり	VPython を用いたシミュレータモデル

6. PA10 システム起動時に立ち上げた RTSystemEditor 上で制御端末 RTC と PA10 システム制御 RTC のポートを接続し、PA10 システム制御 RTC を活性化させる（ポート接続図参照）。

7. PA10 システム起動の手順 1.および 4.を行う。

5 トラブルシューティング

5.1 作業対象物認識率の低下

作業対象物認識に頻繁に失敗するようになってきたり、飲み物の把持点や搭載目標座標がずれるようになってきた場合は、再度カメラ座標キャリブレーションを行う。

6 その他

6.1 特記事項

本書をご利用される場合には、 以下の記載事項・条件にご同意いただいたものとします。

- 本書は独立行政法人 新エネルギー･産業技術総合開発機構の「次世代ロボット知能化技術開発プロジェクト」内実施者向けに評価を目的として提供するものであり、商用利用など他の目的で使用することを禁じます。
- 本書に情報を掲載する際には万全を期していますが、それらの情報の正確性またはお客様にとっての有用性等については一切保証いたしません。
- 利用者が本書を利用することにより生じたいかなる損害についても一切責任を負いません。
- 本書の変更、削除等は、原則として利用者への予告なしに行います。また、止むを得ない事由により公開を中断あるいは中止させていただくことがあります。
- 本書の情報の変更、削除、公開の中断、中止により、利用者に生じたいかなる損害についても一切責任を負いません。
- PA ライブラリは、三菱重工業株式会社の製品であり、権利は三菱重工業株式会社に帰属します。
- API 関数ライブラリ集 API-PAC(W32) は、株式会社コンテックの製品であり、権利は株式会社コンテックに帰属します。
- 高機能 3 次元視覚システム VVV は、産業技術総合研究所が著作権を保持します。

【連絡先】

RTC 再利用技術研究センター

〒101-0021　東京都千代田区外神田 1-18-13　秋葉原ダイビル 1303 号室

Tel/Fax：03-3256-6353　E-Mail：contact@rtc-center.jp